

تو این نوشتار قصد دارم توضیح مختصری در مورد پروژه و مولفه های آن بدهم.

فولدرها

• server

این فولدر شامل فایل های مورد نیاز سرور پروژه است. اپلیکیشن به عنوان کاربر درخواست خود را به سرور ارسال می کند و سرور ارتباط بین اجزای دیگر پروژه را بر عهده دارد و در نهایت نتیجه را به کاربر منعکس میکند. پایگاه داده مورد استفاده در این بخش MongoDB است. سرور این پروژه با Node js آماده شده است.

• App

این فولدر شامل فایل های مربوط به سمت کاربر است. در این بخش کاربر تراکنش مورد نیاز را امضا و ارسال میکند و بقیه اجزا عملیات راستی آزمایی و اجرای آن را بر عهده خواهند داشت. اپلیکیشن با React آماده شده است.

• Processor

این فولدر قلب پروژه محسوب می شود. تراکنش ها در این بخش پیاده سازی می شوند. در واقع قرارداد هوشمند سناریوی شما در این قسمت پیاده سازی خواهد شد.

• ledger_sync

پایگاه داده مورد استفاده در بلاک چین این پلتفرم از نوع (key-value) است که کلید آن یک آدرس ۱۲۸ کاراکتری است. در واقع هر بخش از اطلاعات به صورت کدگذاری شده به عنوان ارزش یک کلید قرار میگیرد. برای استفاده به صورت کارا هر تغییر در این پایگاه داده به صورت یک رویداد منتشر می شود که این بخش آن را تبدیل به پایگاه داده -document based میکند. حال کاربر می تواند با دسترسی به این پایگاه داده جستجوهای خود را انجام دهد. پایگاه داده این بخش با استفاده از RethinkDB است. این بخش هم با JavaScript آماده شده است.

• addressing

در این فولدر آدرس های مورد نیاز برای ثبت اطلاعات در بلاک چین تولید می شود. در نظر داشته باشید که این آدرس ها باید به صورت قطعی باشند و با یک مولفه از داده دارای ارتباط یک به یک باشند. به عنوان مثال کلید عمومی هر کاربر یکتاست و با استفاده از آن آدرس حساب کاربران ساخته می شود.

• protos

این فولدر شامل فایل های مربوط به پروتوهایست. این فایل ها در واقع اسکیم های تراکنش ها و اجزای مربوط به معماری آن هاست، به عنوان مثال اگر برای کاربر حسابی تعریف میکنید در این بخش اسکیم آن مشخص میشود تا تمام اجزا بتوانند اطلاعات را بر اساس آن Encode و Decode کنند.

محتوای قالب پروژه به طور کلی

در این قالب چرخه کامل یک تراکنش پیاده سازی شده است. یعنی از ابتدا که کاربر تراکنش افزایش موجودی حساب خود را میسازد، امضا میکند و سپس به سرور ارسال میکند. سرور تراکنش را به زیر ساخت بلاک چین منتقل میکند، سپس تراکنش در اختیار processor قرار میگیرد. processor بعد از انجام عملیات، نتیجه را به زیرساخت بلاک چین اعلام میکند، در صورت تغییر، پایگاه داده بلاک چین به روزرسانی می شود. به طور همزمان هم با استفاده از ledger_sync پایگاه داده RethinkDB اپدیت میشود و هم نتیجه به کاربر از طریق سرور منتقل می شود.

نحوه اجرا

همان طور که قبلا هم بیان کرده بودم، دانشجویان مجاز هستند با هر زبان و کتابخانه ای که بر آن تسلط دارند کار پروژه را انجام دهند. به عنوان مثال اگر خواستید میتوانید به لینک زیر بروید و فایل های مربوط به پروسسور را با زبان دلخواه کاملاً باز نویسی کنید.

https://sawtooth.hyperledger.org/docs/core/releases/latest/app_developers_guid/using_the_sdks.html

https://sawtooth.hyperledger.org/docs/core/releases/latest/api_references.html

یا اگر نمیخواهید از سرور node js استفاده کنید میتوانید به هر زبان دلخواه آن را پیاده سازی کنید. بقیه قسمت ها هم به همین شکل.

ادامه...

حال شما با توجه به سناریوی طراحی شده توسط خودتان، باید تراکنش ها و اجزای مربوط را طراحی و سپس پیاده سازی نمایید.

- ابتدا در بخش proto تراکنش، اسکیم تراکنش را طراحی و پیاده سازی کنید.
- سپس در بخش processor، عملیاتی که باید توسط تراکنشها انجام شود را پیاده سازی نمایید.
- با انجام تغییرات مورد نیاز در ledger_sync، پایگاه داده بلاک چین و RethinkDB را همگام نگه دارید.
- حال ساخت و امضای تراکنش در بخش کاربر را در اپلیکیشن پیاده سازی نمایید.
- در نهایت برای انجام تراکنش UI (رابط کاربری) مورد نیاز را در بخش اپلیکیشن پیاده سازی کنید.

امیدوارم این نوشتار، بخشی از ابهامات شما را برطرف کرده باشد. برای حل ابهامات احتمالی باقیمانده با من در ارتباط باشید.